THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 61, No. 7, September 1982
Printed in U.S.A.

Stored Program Controlled Network:

NO. 1/1A ESS—SPC Network Capabilities and Signaling Architecture

By L. J. GAWRON, S. J. LUEDERS, and K. L. MOELLER

(Manuscript received June 2, 1981)

The No. 1/1A Electronic Switching System (ESS) will play an important role in the evolving Stored Program Controlled (SPC) Network because, as a major electronic switching system for local and combined local/toll service, it provides direct interfaces to many of the Bell System's customers. It provides important capabilities required for System 800 Service and for a variety of other new SPC network services. This paper describes several of the basic SPC network capabilities provided by the No. 1/1A ESS. It also describes the architecture and implementation of its common-channel interoffice signaling (CCIS) subsystem, CCIS call-processing software, and System 800 software.

I. INTRODUCTION

The No. 1 ESS was the Bell System's first major electronic switching system to provide commercial service. It went into service in 1965 and has served since then as a metropolitan local switching system. It uses Stored Program Control (SPC) capabilities to provide basic telephone service, as well as numerous residential and business features. In 1976, an improved metropolitan local switching system, the No. 1A ESS, went into service. It uses the same switching network as the No. 1 ESS, but with a higher performance processor it has about twice the call capacity of No. 1 ESS. Today, No. 1 and No. 1A ESSs serve nearly one-half of all Bell System subscriber lines. In addition, these systems also provide toll-switching capabilities when they are used as toll offices.

In 1976, SPC toll switching systems were first interconnected with a modern common-channel interoffice signaling (CCIS) system. No. 1 ESS

joined the resulting SPC network with the introduction of CCIS in 1978. No. 1A ESS followed in 1979. The CCIS system currently provides toll network signaling improvements, which result in such benefits as faster call setup, as well as trunk and service circuit cost savings.

Ultimately, the major benefits to be derived from the use of modern common-channel signaling systems will be in the new features and customer services that they can provide. To fully realize these benefits, the SPC network is being extended to include local switching offices which provide direct interfaces with the customer. The No. 1/1A ESSs will be the first local switching systems to utilize CCIS with such service scheduled to begin in 1981. This will then permit local, as well as toll, calls to be handled using CCIS. It will permit basic call-handling improvements such as providing busy tone from the originating office rather than from the terminating office, thus reducing trunk holding time for interoffice calls requiring busy treatment. It will also provide the customer interfaces required for implementation of many new SPC network features and services.

II. NETWORK CAPABILITIES

The SPC network improves basic call handling through the use of CCIS. However, the major benefits of the SPC network will be in the new services made possible because of capabilities provided by SPC network nodes operating in unison. The SPC network plan is described in another article in this issue.[1] The following paragraphs describe SPC network capabilities available in No. 1/1A ESSs.

2.1 Basic signaling

No. 1/1A ESSs have the ability to use CCIS for basic call handling, as well as for special features. This includes a banded signaling capability which uses trunk band and member numbers to identify CCIS trunks in call-processing messages. Banded signaling is described in an earlier article.[2]

The No. 1/1A ESS has the ability to pass along banded messages for an established CCIS trunk connection. This pass-along capability can be used by any switching office in a completely CCIS trunk connection to look ahead or to look backward to an end office for information which can be used in processing the call. For example, a terminating office could use the pass-along capability to transmit a request for calling-party information from the originating office. This information could then be used to provide terminating office services, such as special alerting or special call handling.

The No. 1/1A ESS also has the capability for signaling directly to any other SPC network node. Direct signaling does not require an established trunk connection. One example of its use would be for a

switching office to communicate with a nonswitching office node, such as a centralized data base. Using this capability, switching offices could obtain call-handling information needed for special types of calls. Switching offices, especially local switching offices, could use direct signaling to provide customer status information to such data bases. The data would then be immediately available to the entire SPC network for processing calls. The System 800 features described in Section V of this article use direct signaling for providing customer status information to national data bases and for accessing those data bases to obtain call-handling information. Direct signaling is described in another article in this issue.[3]

2.2 SPC network interfaces

The SPC network provides faster, more efficient basic call handling. It will now also provide a variety of new and improved services. Rapid deployment of these services can be achieved through the use of centralized feature logic and national data bases at SPC network nodes called network control points (NCPs, see Fig. 1). Universal availability of SPC network service depends on all subscribers having access to the SPC network. Access for call handling is provided at SPC network nodes called action points (ACPs). Stored program control local offices, such as No. 1/1A ESSs, could serve as ACPs, thus providing direct SPC network access to customer lines. Customers served by non-SPC local offices can also obtain access to SPC network features via trunks to SPC network toll office or traffic service position system (TSPS) ACPs, as illustrated in Fig. 1. However, such indirect access may involve trunking penalties, call setup delays, and/or overloading at toll offices and TSPSs. Therefore, it is advantageous to provide access to the SPC network at the local office whenever a high percentage of that office's customers subscribe to or are likely to use SPC network services.

In addition to the ACP call-handling interfaces, other SPC network interfaces are required to provide service data to NCPs. Such data might include customer class-of-service information or busy/idle line status. This type of information is often accessible only at the local office. Thus, for certain SPC network services, the subscriber must be served by an SPC network local office. Because No. 1/1A ESSs serve such a high percentage of the subscriber lines including lines to many business customers, they can play a major role in the SPC network.

System 800 will be the first service implemented on the No. 1/1A ESS to utilize the types of SPC network interfaces described above. A busy/idle status indicator (BISI) feature will provide customer line status from local offices to national System 800 data bases in real time. An originating screening office (OSO) feature, operating in either local or toll offices, will access these data bases to obtain call-handling

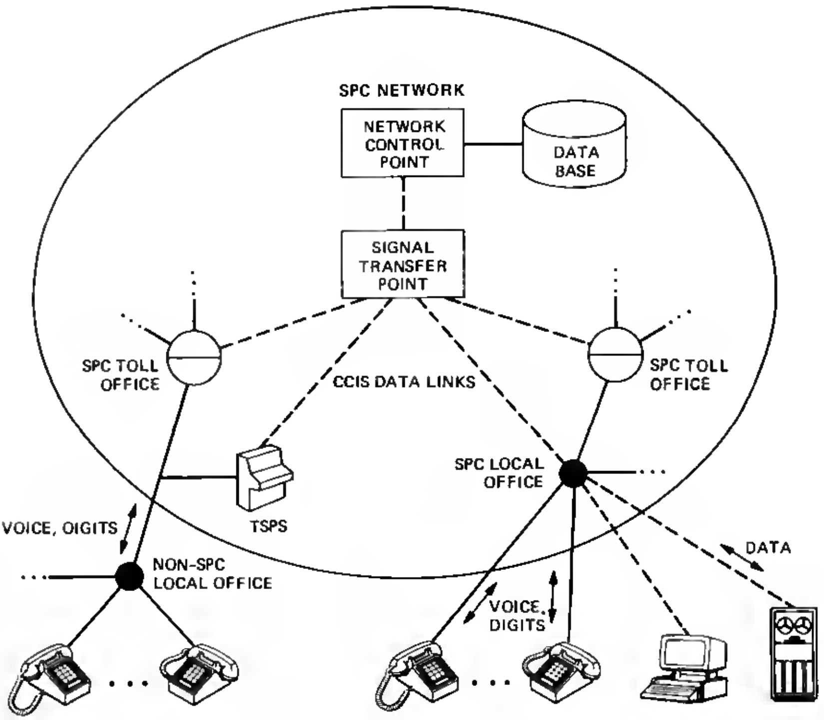

Fig. 1—Stored Program Controlled Network interfaces.

instructions for 800 Service calls. Section V describes the No. 1/1A ESS implementation of these features.

2.3 Customer Interfaces

The objective of the SPC network is to provide new and improved customer services. Some services can be provided by an intelligent network which contains a variety of customer data stored either in centralized data bases or in local switching offices. Other services will require new customer interfaces including new dialing sequences and voice dialogues for prompting customers and providing call status. These interfaces are part of the SPC network dialing plan described in this issue.[4] Some services may also require nonvoice interfaces for communicating service-related information between the SPC network and the customer. The following paragraphs describe some of the useful customer interface capabilities of No. 1/1A ESS local offices.

The SPC local offices have direct access to subscriber lines and to line-related data stored within the office. Line access allows local offices to collect service-related data through special dialing sequences

from customers using either rotary dial or *Touch-Tone** telephones. Toll offices may also collect digits via trunk connections, but this is only practical from *Touch-Tone* phones. The TSPS may also collect such information from customers verbally.

Direct line access permits special alerting, such as distinctive ringing. Since alerting is strictly a local office function, special alerting for SPC network services must be provided from SPC network local offices.

Access to line-related data contained within local offices allows these offices to provide such data to other SPC network nodes when required for special features. For example, an originating office could provide calling number information to an NCP as part of a call-processing query or to a terminating office in response to a pass-along request for such information. Local offices could provide busy/idle line status either on request or whenever line status changes. They could also provide customer class-of-service information indicating the types of services that customers have subscribed to or currently have active.

In addition to line-related customer interfaces, SPC network features may require special data interfaces with the customer. As illustrated in Fig. 1, the No. 1/1A ESS could provide such interfaces to customer computers or keyboard terminals. These interfaces could be used for exchanging service control and/or status information. The No. 1/1A ESS currently provides such interfaces to private network customers and System 800 customers.

III. SIGNALING SUBSYSTEM ARCHITECTURE

3.1 Objectives

The objectives of the No. 1/1A ESS signaling subsystem include providing hardware-independent signaling capabilities to a variety of No. 1/1A ESS application programs. These include programs such as CCIS call processing and System 800 features that use CCIS directly. They also include many other feature programs that use CCIS indirectly, e.g., via general-purpose trunk supervisory programs.

Because toll offices handle greater volumes of interoffice calls, they generally have greater signaling capacity requirements than local offices. However, local offices have more stringent data-link equipment cost constraints than toll offices. Two different data-link hardware subsystems have been developed to meet these differing needs in No. 1/1A ESS offices. Certain software must be included in a switching office when a particular data-link subsystem is used. Other software is common to both subsystems. Software packaging must be provided

* Registered service mark of AT&T.

that allows an office to load only the software required for its particular equipment arrangement.

The following sections describe the signaling subsystem architecture that satisfies these objectives.

3.2 General description

The No. 1/1A signaling subsystem comprises several layers of control as illustrated in Fig. 2. This structure allows ESS applications software to send and receive CCIS messages without being concerned about Input/Output (I/O) details or data-link administration procedures.

The ESS applications software that uses the signaling subsystem includes call processing, maintenance, network management, and other feature programs. Supervisory programs provide special signaling interfaces for application programs that perform trunk-related signaling without knowledge of whether a trunk is a CCIS trunk or not.

The signaling software layer provides hardware-independent signaling capabilities. It provides macro-accessible subroutines for formatting and transmitting CCIS messages. It also reads CCIS messages from the data-link equipment and delivers them to appropriate ESS applications programs.

The data-link software layer provides all hardware-dependent inter-

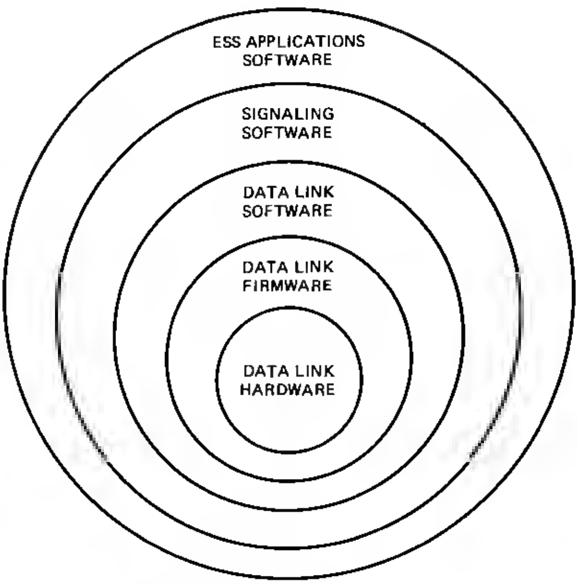

Fig. 2—Signaling subsystem layers of control.

faces for each type of CCIS data-link equipment. It also provides data-link administration and maintenance capabilities.

Each of the CCIS data-link hardware subsystems used in No. 1/1A ESS is microprocessor controlled. The data-link hardware, along with its firmware programs, implements much of the CCIS protocol including message queuing, data transmission, error detection, and error correction through retransmission. The firmware also includes diagnostic programs that can be invoked by data-link layer software.

The following paragraphs further describe each of the signaling subsystem layers beginning with the lowest layer—the data-link hardware.

3.3 Hardware

The No. 1/1A ESS currently supports two types of CCIS data-link hardware. One is a 2400 b/s data link called the CCIS data terminal frame (DTF). It is the result of a common development for initial CCIS service on CCIS signal transfer points (STPS), No. 4A/Electronic Tandem Switching (ETS), No. 4 ESS, and No. 1/1A ESS toll switching offices. The CCIS-DTF satisfies the common needs of toll network STPS and CCIS switching offices, i.e., high volumes of trunk-related signaling traffic. The CCIS DTF is used only for CCIS signaling.

The other type of data link used for CCIS is the peripheral unit controller-data link (PUC-DL). The peripheral unit controller (PUC) was developed to provide microprocessor control of digital carrier trunks (DCTS). The PUC was later modified to provide microprocessor control for data-link applications. The resulting PUC-DL was first used to communicate with a remote switching system (RSS) and was later adapted for ETS private network service and then for CCIS service. A single PUC-DL frame can be shared among all three applications. Each application has its own type of data terminal or line interface unit. The CCIS PUC-DL terminal, like the CCIS-DTF terminal, operates at 2400 b/s.

Because of several years advance in technology between development of the CCIS-DTF and the PUC-DL, the PUC-DL is not only smaller than the CCIS-DTF but its per-terminal cost is less than the CCIS-DTF per-terminal cost. The cost advantage of a CCIS PUC-DL terminal is increased when the PUC-DL frame is shared with other applications in the same office. Thus, the PUC-DL is not only a cost-effective terminal for local offices, but it can be used as a cost-reduced terminal for No. 1/1A ESS toll offices as well.

Figure 3 illustrates the No. 1/1A ESS's signaling subsystem architecture used for CCIS. This illustration is applicable to both the CCIS-DTF and the CCIS PUC-DL. The data-link equipment provides the interface

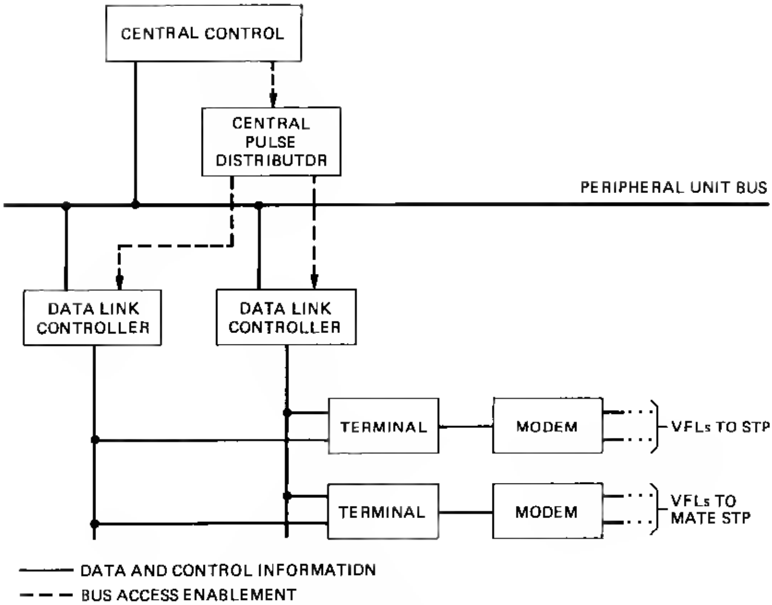

Fig. 3—Signaling subsystem hardware.

between the ESS central control and the Voice Frequency Links (VFLS) used for data transmission.

Each type of signaling subsystem used for CCIS comprises a duplex data-link controller and one or more pairs of data-link terminals. The CCIS data links are always assigned in pairs, with the number of pairs determined by the volume of signaling traffic in the office. Each link of a pair connects the ESS to a CCIS STP. Signaling traffic is shared on each link of the pair, both of which are concurrently active. Each link includes two geographically diverse VFLS—one active and one standby. This permits rapid link recovery in the event of transmission failures. The signaling network is described in greater detail in an earlier article.[5]

Data and control information is exchanged between the central control and the signaling subsystem via the peripheral unit bus. Many peripherals are connected to the bus. Thus, the central control uses a central pulse distributor to enable a particular peripheral to access the bus when there is data or control information on the bus for it.

Both the CCIS-DTF and the PUC-DL provide microprocessor-controlled, self-diagnosing data links. The CCIS-DTF has a hard-wired controller and a microprocessor-controlled terminal. The CCIS-DTF terminal's program is loaded from the central control when it is initialized. The PUC-DL has a microprocessor-controlled data-link con-

troller and terminal, each of which contains permanently resident programs. The functions performed by the microprocessor programs are described in Section 3.4.

The data-link hardware provides electrical interfaces and low-level signaling operations necessary to implement the CCIS protocol. These operations include parallel-to-serial conversion of CCIS signal units transmitted over the electrical interface between the data terminal and the modem and serial-to-parallel conversion of signal units received over this interface. The modems perform digital-to-analog and analog-to-digital conversion of the transmitted- and received-bit streams. They also maintain bit synchronization on the VFL.

3.4 Firmware

The firmware is the collection of programs that reside in the data-link hardware. The firmware implements the link-level CCIS protocol.* This level provides error-free data transmission over a signaling link in the CCIS network.

The CCIS-DTF and CCIS PUC-DL firmware formats CCIS signal units and messages into fixed-length blocks for transmission over the link. The firmware generates a cyclic redundancy check (CRC) code† used for detecting transmission errors on each signal unit, and it retransmits messages containing signal units received in error at the other end of the link.

The firmware provides priority-level queuing of messages waiting to be transmitted on the link and of messages received on the link that are waiting to be unloaded by the central control software. It buffers all transmitted messages until they are acknowledged so that they are available for retransmission if required. It also provides positive acknowledgment of all signal units received correctly and negative acknowledgment of all signal units received in error on the link.

The firmware collects message traffic counts and error statistics, such as the number of address messages transmitted and the number of signal units received in error. This information is provided to the central control software upon request. The firmware detects and reports problems such as circuit failures, buffer overflows, and exceeded-error thresholds. The firmware also responds to software requests for data-link control and maintenance actions such as initialization, reconfiguration, and diagnosis.

3.5 Data-link software

The data-link software layer is responsible for maintaining an op-

---

* For a description of the CCIS system protocol, see Ref. 2.

† In the PUC-DL, the firmware computes the CRC code. In the CCIS-DTF, the hardware computes the CRC code under control of the firmware.

erational signaling subsystem. It is also responsible for providing signaling subsystem status and hardware-dependent I/O interfaces to the outer layers of software illustrated in Fig. 2. The major components of this layer are the link security and maintenance software. There are CCIS-DTF and PUC-DL programs in each of these categories.

Link security software provides CCIS data-link administration, which includes fault detection and automatic link recovery procedures. The objective of these procedures is to quickly remove faulty link components from service and to maintain an operational signaling subsystem by providing an alternate path for affected signaling traffic.

Link security maintains ESS signaling subsystem status and signaling network status for use in output message routing by the signaling software layer. This includes status reflecting the operational states of the data-link equipment in the ESS office. It also includes status of the operational states of other links in the signaling network that affect the routing of trunk-related (banded) messages that emanate from the ESS. Such signaling network status is received in messages from the STPS.

One of the principal link security procedures for maintaining an operational signaling subsystem is automatic link recovery. This procedure is initiated when link security detects a service affecting data-link fault or alarm. Since CCIS links are always equipped in pairs, the link recovery procedure begins by setting the failed link's operational status to out of service and placing the link in a faulty link mode of operation. This immediately causes banded messages destined for the failed link to be routed to its mate link and direct-signaling messages to be distributed over all other available links. It also causes change-over signals to be transmitted on the failed link, if possible, in order to notify the STP of the failure in the event that the STP had not already detected it. Link security then transfers all messages awaiting transmission or retransmission and unacknowledged transmitted messages from the failed link to its mate.

The part of the recovery procedure described above immediately restores the lost signaling capability. The next step in the procedure is to restore the lost link as quickly as possible. Since the most common source of link failure is the VFL, each CCIS link has a duplicate VFL as illustrated in Fig. 3. When both ends of a link have detected a failure, they will attempt to resynchronize on the backup VFL. If that attempt is unsuccessful, they will alternately attempt to resynchronize on the original VFL and then again on the backup VFL. The ESS alternates between VFLS every 5 seconds, and the STP alternates between VFLS every 10 seconds to guarantee intervals when both ends are attempting to resynchronize on the same VFL. In most cases, this procedure restores the link. Following a successful resynchronization on either

VFL, a 15-second prove-in period is entered to verify acceptable transmission quality on the link. Following a successful prove-in period, the link security software notifies the STP at the other end of the link that signaling traffic may be returned to the link by transmitting a load transfer signal. When the STP returns a load transfer acknowledgment signal, link security allows the ESS to resume signaling on the link by setting the link status to active. If link resynchronization is unsuccessful after 3 minutes of attempting, link security requests maintenance programs to diagnose the affected data-link equipment.

The software maintenance programs control execution of data-link diagnostic programs that reside within the data-link equipment. The diagnostic programs verify access to different points within the data-link equipment, and they also execute data-link equipment tests. Diagnostics can be requested by link security software or manually via a teletypewriter. The maintenance programs also support manual controls for data-link configuration and testing. For example, they respond to requests to remove data links from service, to switch VFLs, and to provide maintenance access to the VFLs for manual testing. These maintenance functions are done in conjunction with the link security software.

3.6 Signaling software

The signaling software layer provides hardware-independent signaling capabilities to the ESS applications software. This includes I/O interfaces for CCIS trunk-related (banded) signaling and direct signaling. The signaling software also provides automatic responses to signaling network congestion. The principal programs in this layer are a CCIS input processor, a CCIS output processor, and signaling network congestion control programs.

The CCIS input processor is a continuous process within the ESS central control. It executes at regular intervals, each time checking for input from the CCIS data links. When input messages are present, the input processor unloads the messages from the data links. Within the programs that unload the data links, there are short hardware-dependent segments of program code that are both logically and physically part of the data-link software layer. Each program or program segment resides in a software feature package. This concept is explained in Section 3.8.

The input processor distributes CCIS input messages to appropriate application programs. Included in the input processor is a finite-state machine (FSM) controller that provides inputs to application programs such as CCIS call processing, which uses an FSM-based software architecture. For these applications, the specific program that receives an input message is a function of the current state of the FSM at the time

the input is received. The FSM controller also performs state updates when the application program has completed processing of the input message. The call processing FSM is described in Section 4.3.

The CCIS output processor consists of macro-accessible subroutines which are called by the application programs when they wish to send a CCIS message to a different SPC network node. The output processor provides a high-level interface that permits the application programs to be concerned mainly about the application-oriented data content of messages and not about the low-level signaling protocol characteristics.

The output processor will optionally format user-specified data into CCIS message format. It then routes the messages to an appropriate signaling link based on data-link equipment status and signaling network status maintained by the data-link software layer and also based on a terminal load-balancing algorithm. The load-balancing algorithm strives to maintain an evenly balanced load among all available CCIS links. Banded messages are associated with a specific data-link pair; thus, such messages are evenly distributed to the two terminals of the pair. Direct-signaling messages may be routed over any available CCIS link; therefore, these messages balance the signaling load across terminal pairs. The output processor calls data-link-dependent output subroutines in the data-link software layer to transmit data. These subroutines execute peripheral orders to effect the transfer of data across the peripheral unit bus to the data-link equipment.

Signaling network congestion control programs automatically respond to signaling overload in the ESS data links and in the CCIS network STPs. Data-link overloads are detected as buffer-full conditions in the data links. Overloads in STPs directly connected to the ESS are reported to the ESS using processor signaling congestion messages. Because of the load-balancing algorithms used within the signaling network, it is assumed that whenever a data link or STP becomes overloaded that its mate is also overloaded. Thus, the response to either of these conditions is to reduce traffic to both of the affected links while such congestion lasts. Signaling traffic is reduced by preventing new call originations which would use the affected terminal pair and by preventing direct-signaling traffic from using that terminal pair. Overloads in STPs not directly connected to a particular ESS can still affect signaling for specific trunks in that ESS whose CCIS messages must be routed through the overloaded STPs. The ESS is notified of such overloads through group signaling congestion messages. These messages apply to specific trunk groups. The ESS response to one of these messages is to place a 10-second network management trunk group control on the affected group. This temporarily suspends signaling traffic for that trunk group by either canceling new call originations destined for the group or optionally skipping the group in the

call-routing sequence and permitting the call to complete on a different trunk group.

3.7 ESS applications software

The applications software consists of ESS programs that use the signaling subsystem for basic call handling, for implementing special features, etc. Two principal applications are CCIS call processing and System 800. These applications are described in Sections IV and V, respectively. There are also other important applications which use CCIS. For example, CCIS network management provides traffic controls on the CCIS portion of the trunking network. Trunk maintenance programs in switching offices at each end of CCIS trunks use the signaling network to exchange maintenance state information about the trunks. Trunk query is an interoffice trunk state audit which verifies that the operational and/or maintenance states at each end of CCIS trunks are consistent. Other ESS call-processing programs use the signaling subsystem for trunk-related signaling via the supervisory program interfaces described in Section 4.2.

In addition to application layer programs, signaling software and data-link software layer programs themselves use the signaling subsystem.

3.8 Software feature packaging

No. 1/1A ESSS serve a variety of purposes. They provide basic local, tandem, and toll switching, as well as a variety of network features such as CCIS and customer services such as System 800. Each office is individually engineered and equipped with the hardware and software required for the features and services provided by that office. Feature packaging is a mechanism used to permit an office to load that software, and only that software, needed to provide the office's particular combination of features and services.

A feature package is a collection of software associated with a particular feature, service, or piece of equipment. The software may include complete programs, program segments, and/or subroutines. It can also include fixed and variable amounts of temporary (call store) memory. A feature, service, or piece of equipment may require one or more feature packages to be loaded into an office. Also, a particular feature package may be used by one or more different features, services, or pieces of equipment. The following paragraphs describe the feature packages which provide CCIS capabilities in local and toll offices, the packages associated with the signaling system hardware, and the System 800 packages.

CCIS Common—This package contains software required by every

No. 1/1A ESS office equipped with the signaling capability. It includes hardware-independent signaling software and CCIS applications software, such as call-processing programs, which are common to local, tandem, and toll switching, trunk maintenance, network management, etc.

Local CCIS—This package contains CCIS call-processing logic required in local offices. It handles calls routed over CCIS interlocal, tandem, and toll-connecting trunks.

Toll CCIS—This package contains CCIS call-processing logic required in toll offices. It handles calls routed over CCIS intertoll and toll-connecting trunks.

CCIS Two-Wire—This package contains maintenance programs required to diagnose faults in two-wire CCIS trunks and continuity check circuits used with CCIS trunks. Two-wire networks are used mainly for local switching.

CCIS HILO—This package contains maintenance programs required to diagnose faults in HILO CCIS trunks and continuity check circuits. The HILO switching networks provide transmission quality equivalent to four-wire networks and are used for toll switching in No. 1/1A ESS offices.

2400DL—This package contains maintenance software for the 2400 b/s CCIS-DTF.

CCIS 2400DL—This package contains hardware-dependent I/O interfaces and link security logic associated with the CCIS-DTF.

PUC—This package contains maintenance software associated with the peripheral unit controller. This software is common to the DCT application and also to each of the PUC-DL applications.

PUC-DL—This package contains maintenance software associated with the PUC when it is used as a data-link controller. The software is required for any application that uses the PUC-DL.

CCIS PUC-DL—This package contains hardware-dependent I/O interfaces, link security logic, and maintenance programs associated with the CCIS configuration of the PUC-DL.

OSO—This package contains the software which comprises the System 800 originating screening office feature. It is used in all No. 1/1A ESS local and toll offices that have the CCIS signaling capability.

BISI—This package contains the software which comprises the System 800 busy/idle status indicator feature. It is used in local offices that serve System 800 customers.

The required combination of the above feature packages depends on the type of office, the data-link hardware subsystem used, and the features or services provided by that office. Figure 4 illustrates the feature package combination required for a typical local switching office with CCIS. It uses PUC-DL for its CCIS data links. It provides

originating office screening of 800 Service calls, and it serves customers that have 800 Service lines connected to the office.

Every No. 1/1A ESS has a collection of base software which provides basic call processing, maintenance, and switching office administration. Feature packages usually build on capabilities provided by the base software. As illustrated in Fig. 4, feature packages can also build on the capabilities provided by other feature packages. For example, the CCIS PUC-DL package adds data-link capabilities which are unique to CCIS onto the more general PUC-DL capabilities provided by the PUC-DL package. The PUC-DL package adds general data-link capabilities to the more general PUC capabilities provided by the PUC package. The PUC package, in turn, builds on basic maintenance capabilities provided by the ESS base software. Likewise, CCIS applications such as OSO and BISI use the signaling capabilities provided by the CCIS common package in order to implement their own features.

IV. CCIS CALL PROCESSING

4.1 Development environment

CCIS is sometimes referred to as a new signaling type to replace multifrequency (MF) signaling. While this may be partly true, address signaling (digit transmission) is certainly not the entire concept, nor the most difficult aspect, for call processing to implement. Digit transmission and voice path assurance for CCIS involve only a few messages and, as with MF signaling, comprise only the addressing portion of the call. It is the expanded capabilities of CCIS (e.g., backward call failure messages) that make CCIS difficult to integrate into existing systems which before only dealt with in-band signaling and supervision (e.g., trunk on-hook/off-hook signals).

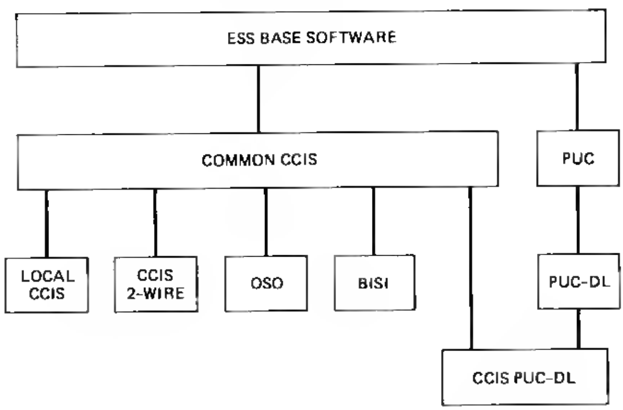

Fig. 4—Software feature packaging for an SPC network local office.

The CCIS call processing is involved in setup and teardown of calls utilizing CCIS trunks and mainly handles CCIS trunk-related (banded) messages. This processing involves supervisory messages such as trunk seizure, answer, and disconnect which have non-CCIS trunk on-hook/off-hook signaling counterparts. However, CCIS call processing also involves new nonsupervisory information messages such as specific call failure, trunk reset, and address messages which may not have non-CCIS trunk signaling counterparts.

The CCIS call processing is faced with other issues not previously dealt with by existing call processing. Because of signal transmission errors and resultant retransmission, CCIS signals can arrive out of sequence, be received twice, or even spill over from previous calls. Unexpected signals can arrive at any time, and special timing may be required to determine their reasonableness.

No. 1/1A ESS first developed CCIS for the toll environment. It was decided then that because of the number of signals and internal inputs to be processed during all phases of the calls, a finite-state machine structure would be best suited for the implementation. Also, since the toll environment is very limited in the types of connections required (i.e., trunk-to-trunk only), the processing logic could be mostly separate from the existing toll-processing logic. This eliminated extensive changes in the existing toll-call processing logic to handle supervision, out-of-sequence messages, new nonsupervisory messages, etc. As soon as the call-processing logic determines that a call involves a CCIS trunk, control is passed to the toll CCIS logic which then maintains control of the call through disconnect processing.

Local CCIS development was faced with an even larger and more complex environment. The local call-processing environment involves many program interfaces and types of connections requiring hundreds of thousands of program instructions. For example, besides basic calls such as line-to-trunk and trunk-to-trunk, there are interfaces with features such as coin, three-way calling, call waiting, etc. Duplicating this logic to process local CCIS calls would have been too costly initially and most likely would have incurred huge maintenance costs. Thus, it was clear that the existing logic had to process local CCIS calls and, further, coexist with the toll CCIS logic already deployed. Supervisory messages received would have to be converted to their on-hook/off-hook counterparts and passed through some interface to the existing logic. Similarly, trunk circuit state changes to send supervisory signals outward would require conversion to CCIS messages. Nonsupervisory messages (i.e., those with no on-hook/off-hook equivalent) would require special handling apart from the existing logic. This special handling would have to be confined to areas where signaling-type differences are recognized.

4.2 Supervision modernization

The call-processing logic which handles calls involving in-band signaling trunks must also be used to process calls involving local CCIS trunks. To accomplish this, the mechanisms for controlling supervision had to be changed. No longer could the application logic assume on-hook/off-hook supervision on the trunks. Application programs could no longer directly access I/O memory to change supervisory status. Turning off scanning of a trunk would not necessarily suspend supervision, nor would changing the supervisory state in a trunk circuit necessarily send a supervisory signal to the connected office. An interface between the application programs and the signal I/O programs to isolate supervision had to be developed.

The incoming signal control mechanism which existed had two levels of programs that communicated using shared memory. The I/O level programs detected trunk supervisory changes, updated I/O memory, and reported the changes to the application program level. Application programs controlled report generation by writing directly into I/O memory. The application programs had to be familiar with the use of the memory and the synchronization problems which arose from multiple access by both levels.

The objective of supervision modernization was to eliminate the tight coupling that existed among application programs, I/O processes, and I/O memory. Incoming signal control programs provide the primary interface between I/O and application programs. The application programs control which reports are generated by making requests to the supervisory control program. This control program accesses I/O memory when necessary. Knowledge of I/O memory use and synchronization problems is thereby confined to the signal-processing interface programs. These supervisory interface programs maintain per trunk supervisory status in dedicated memory. Incoming signals from CCIS trunks are recorded in this memory and delivered as logical reports to the call-processing programs. Implementation of this objective effectively isolates the details of supervision from the call-processing programs which allows CCIS and non-CCIS trunks to be treated identically by the call-processing software.

A second supervisory interface, the outgoing signal function, is used to generate outgoing supervisory signals. This interface is used whenever call processing is attempting to change the supervisory state of an interoffice trunk which could possibly be a CCIS trunk. The outgoing signal function allows the call-processing program to specify the logical supervisory message it wants to send and the trunk for which it should be sent. The outgoing signal interface causes the CCIS signaling logic to send the appropriate CCIS supervisory message for a CCIS trunk.

4.3 Structure

The CCIS call-processing logic structure is based on finite-state machines. Each state represents a condition or processing stage of a CCIS trunk involved in a call. The state of each CCIS trunk is stored in a per-trunk state word block of memory. The states are grouped into models which reflect call functions (e.g., continuity checking). All stimuli handled by each model come as inputs through the CCIS input processor where validity screening and state table execution take place.

The CCIS messages are unloaded from the data links by the input processor as described in Section 3.6. The trunk label from a call-processing message is translated to a trunk network appearance and a state word memory address. The message input is applied to the current state which specifies the model in control. The model controls all processing by calling any number of closed transition routines (i.e., those which return to the calling program), updating the state, and possibly calling an open interface routine (i.e., one which allows processing to pass to other application logic).

Models are of two basic types: processing models and conversion models (see Fig. 5). Processing models handle the parts of call processing which are unique to the signaling type. The CCIS address message processing and trunk-continuity checking are the primary examples. They have in-band signaling counterparts that are handled by separate programs. During these portions of a call, the processing models act as application programs and perform call-processing functions. These models are in control prior to sending or receiving the

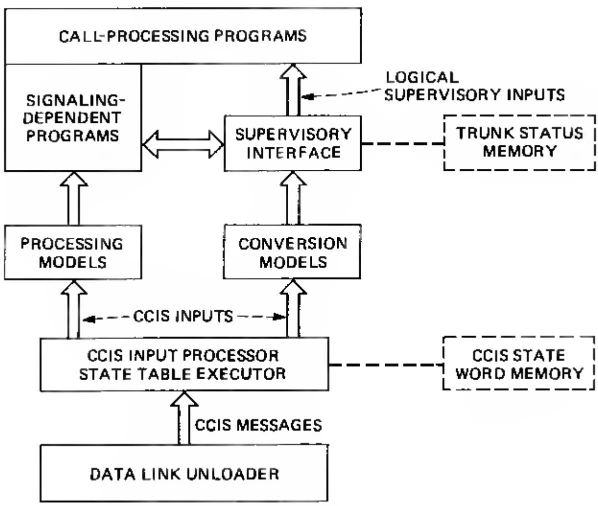

Fig. 5—Local CCIS call-processing structure.

address complete (ADC) message. The ADC message basically separates call setup from answer and disconnect processing. Since most nonsupervisory messages are exchanged prior to ADC, the processing of these messages is confined to the CCIS application programs. All supervisory and nonsupervisory inputs are handled directly by these models.

Conversion models are in control when the only messages to be processed are the supervisory messages. These models validate and resequence messages if necessary before passing them through the supervisory incoming signal interface to the call-processing programs. They also receive the stimulus from the outgoing signal interface to send the appropriate CCIS supervisory message. These models maintain trunk states for message-screening purposes only and provide no call-processing functions themselves. In essence, the states of these models are unaware of whether the CCIS trunk is connected to a line or to another trunk. This terminal processing concept enables local CCIS to interact with the majority of the application programs of the local ESS environment.

4.4 Call flow

The basic call types described in this section are those of the local office. A brief call flow of each type is given specifying the basic program interactions and message protocol. The toll office call-type descriptions have been previously given in other articles and are not repeated here for No. 1/1A ESS.[6]

4.4.1 Originating outgoing calls

An originating outgoing CCIS call begins the same as any originating outgoing call. Dial tone is given to the customer, digits are collected and analyzed, and an outgoing route is selected. When this outgoing route involves a CCIS trunk, the CCIS outpulsing logic receives control. The CCIS outpulsing finite-state model and associated transition routines perform the necessary call-processing functions and remain in control until the receipt of the ADC message. These functions include sending the initial address message (IAM), performing the continuity check if necessary, sending the continuity (COT) message, and handling any backward failure messages. Once ADC has been received or an error has occurred, control is returned to the call-processing programs to perform call setup or teardown as required.

During these and subsequent call functions, a conversion model handles the supervisory message inputs. The answer charge (ANC) message, for example, causes the conversion model to pass a logical answer via the supervisory incoming signal interface to the call-processing programs where answer processing occurs. Upon receipt of disconnect, the call-processing programs tear down the cross office

path and restore the outgoing trunk to the idle state. Here, interface logic for CCIS takes control of the outgoing trunk, sends a clear forward (CLF) message, and places the trunk in a processing model which waits for the release guard (RLG) message. When RLG is received, the trunk is returned to the idle state and made available for another call.

4.4.2 Incoming terminating calls

This call begins with the receipt of an IAM for a particular incoming CCIS trunk. A continuity check circuit is connected to the incoming trunk, if required, and the call waits for the COT message. This function is handled by the CCIS incoming call-processing model as a unique inpulsing-type application program which remains in control until the COT message is received. Upon receipt of the COT message, the ADC message is sent to the originating office. Then control is given to the call-processing programs to perform digit analysis and establish necessary connections. Giving up control means that a conversion model handles subsequent inputs. However, in the event that digit analysis determines that the terminating line is busy, the subscriber busy (SSB) message is returned in lieu of the ADC message so that the customer can be connected to busy tone in the originating office. If the called line is idle, the CCIS incoming trunk is connected to audible ringing tone and ringing is applied to the terminating line. The calling customer then waits for the called party to answer.

If the called party answers, the call-processing logic sets up the cross office path and puts the incoming trunk in the off-hook state. At this point, an outgoing signal interface to send logical answer allows the conversion model to send the ANC message. Upon receipt of a CLF message (i.e., calling party disconnect), the model passes a logical disconnect via the supervisory incoming signal interface to the call-processing programs. These programs tear down the cross office path and restore the incoming trunk to the idle state. An interface for CCIS in the trunk idle logic sends the RLG message, and the trunk is made available for another call.

4.4.3 Tandem calls

An incoming call which routes back out of the office on an outgoing trunk is a tandem call. This call proceeds as an incoming terminating call, except that digit analysis dictates that an outgoing trunk be selected. When the outgoing trunk is a CCIS trunk, the CCIS outpulsing logic receives control to handle the outgoing trunk processing as described in originating outgoing calls, Section 4.4.1. The tandem call-processing logic performs the call setup and teardown processing. Each CCIS trunk is independently associated with a finite-state model. The

models separately handle the CCIS messages and supervisory interfaces for each trunk.

V. NO. 1/1A ROLE IN SYSTEM 800

No. 1/1A ESS performs two important functions in System 800: the OSO function and the BISI function. The OSO function can exist in both No. 1/1A ESS local and toll offices. An OSO queries an NCP on all 800 Service calls to obtain a direct distance dialing (DDD) number for routing. The BISI feature, which monitors the busy/idle status of 800 Service customers and reports changes in status to an NCP, can reside in No. 1/1A ESS local offices. The busy/idle status can then be used by the NCP to provide alternate handling of 800 Service calls that would have received busy treatment. An overall description of the System 800 capability is provided in another article in this issue.[7] The following sections describe the No. 1/1A ESS implementation of the OSO and BISI features.

5.1 Originating screening office

The OSO feature provides single-number DDD calling and improved routing for 800 Service calls. This section discusses the functioning of a No. 1/1A ESS OSO as it applies to both local and toll offices.

5.1.1 Processing 800 service calls

A No. 1/1A ESS OSO processes both originating 800 Service calls and 800 Service calls which arrive over a trunk from a distant office. The 800 Service calls are recognized by examining the first three digits dialed by the customer or received from the distant office. After identifying a call as 800 Service, the OSO determines the identity of the originating Numbering Plan Area (NPA).

The NPA and the dialed 800 number are then formatted into an 800 Service direct-signaling message called QUERY which is sent to an NCP. The NPA is used by the NCP to determine if the call is allowed, based on the customer's purchased service area. The 800 Service call is suspended until a REPLY message is received from the NCP. The OSO saves the call data, while the QUERY is being processed by the NCP, and times for a response from the NCP.

When the OSO receives a REPLY from the NCP, it first determines which call the REPLY is associated with by accessing the saved call data. If the REPLY contains a 10-digit DDD number, the OSO routes the call as a normal DDD call. If a DDD number is not contained in the REPLY (e.g., because of all 800 Service lines being busy), the OSO routes the call locally to an appropriate tone or announcement.

The OSO also has the ability to manually test the OSO-NCP interface. The input of an 800 number and an NPA from a teletypewriter will

cause the ESS to send a QUERY to the NCP. The information contained in the resulting REPLY is then displayed on a teletypewriter at the OSO.

5.1.2 800 Service code controls

The 800 Service code controls are initiated automatically to prevent overloading the switching network, the CCIS signaling network, or the NCPs. Code controls restrict the number of queries sent to a particular NCP.

There are two types of 800 Service code controls:

(*i*) Network management-code controls—These controls can be initiated automatically by the NCP or manually at the OSO. When they are in effect, the OSO limits the number of queries sent to the NCP for a particular 800 number or for a group of 800 numbers.

(*ii*) CCIS failure-code controls—These are initiated automatically when the OSO is unable to successfully send a QUERY. They cause the OSO to restrict queries for a group of 800 numbers.

5.2 Busy/Idle status indicator feature

The BISI feature resides in No. 1/1A ESS 800 service terminating-end offices (TEOs). An 800 Service TEO is a local office which serves 800 Service customers. The BISI feature monitors the busy/idle status of 800 Service customer line groups and reports changes in busy/idle status to an NCP. The NCP can then use the busy/idle status information to provide alternate handling to 800 Service calls which would have received busy treatment.

5.2.1 Busy/idle monitoring and reporting

With the introduction of System 800, all 800 Service calls will be routed to the TEO using a 10-digit DDD number. In a No. 1/1A ESS TEO, each 800 Service DDD number has an associated Simulated Facilities Group (SFG). The SFG is used to limit the number of simultaneous calls to a group of physical 800 Service lines and to provide proper 800 Service band hunting and billing. An 800 Service band is a geographical area from which an 800 Service customer may receive calls. The DDD number also directs the incoming call to the correct physical facilities (i.e., lines).

An SFG is required for each service-area band that an 800 Service customer purchases, so that incoming calls can be billed based on band. Figure 6 illustrates an 800 Service customer setup at the TEO. This customer has seven physical lines and wishes to allow at most five of these lines to be used at any given time for incoming 800 Service calls. The customer wishes to permit three simultaneous calls from band 1 and two simultaneous calls from band 2, and to have band 1

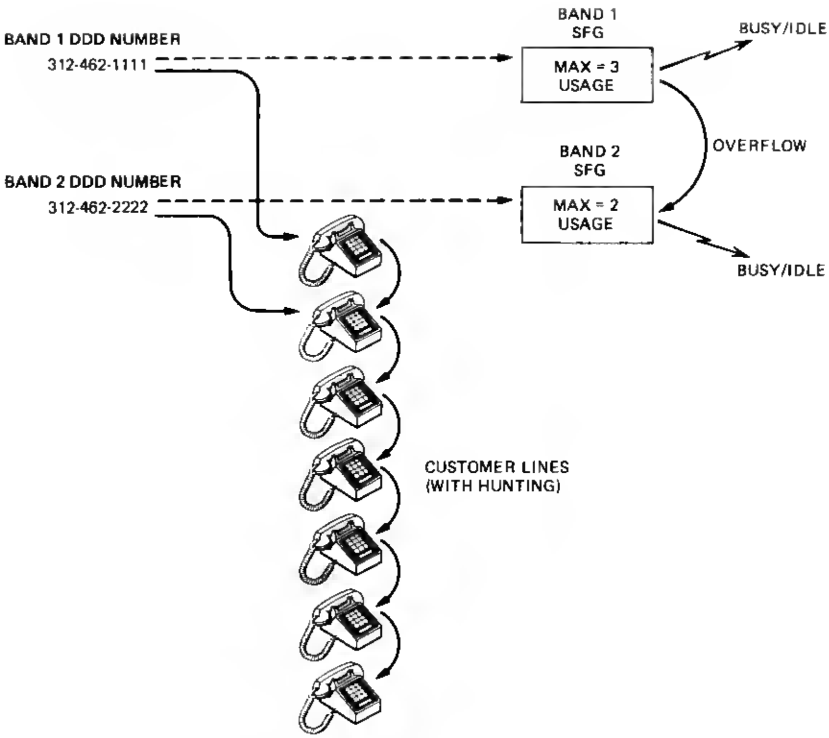

Fig. 6—800 Service customer line group arrangement.

calls overflow to band 2 lines if all band 1 lines are busy. In this case, two SFGs are required, and the band 1 SFG overflows to the band 2 SFG. There are three 800 Service lines in the band 1 SFG and two 800 Service lines in the band 2 SFG. A distinct ten-digit DDD number is required for each band.

The BISI feature monitors the busy/idle status of the 800 Service SFGs in a TEO. To perform the monitoring function, a count of the number of calls currently in progress for each SFG is maintained. This count is called USAGE. Also, for each SFG, a constant is stored which indicates the number of 800 Service lines for that SFG. This per-SFG constant is called MAX. The BISI feature monitors the activity on the SFG and increments or decrements USAGE when 800 Service lines are seized or released.

The BISI feature reports changes in SFG busy/idle status to a remote NCP via CCIS direct-signaling messages. Therefore, if USAGE becomes equal to MAX when an 800 Service line is seized, or if an 800 Service call is unable to complete because USAGE=MAX, a direct-signaling message is sent to the NCP that indicates BUSY. If USAGE becomes

equal to MAX-1 when an 800 Service line is released, an IDLE direct-signaling message is sent. The TEO continuously audits USAGE so that its value is correct at all times.

To assure that the BUSY and IDLE messages for a particular SFG are properly sequenced, the TEO performs a blind-period timing function. An SFG goes on blind period timing after a BUSY direct-signaling message is sent. While an SFG is on blind-period timing, no BUSY or IDLE direct-signaling messages are sent to the NCP. When the blind-period timing interval expires, an IDLE direct-signaling message is sent if the busy/idle status changed to idle while the SFG was on timing.

The NCP has the ability to query the TEO about the current busy/idle status of a particular SFG. When the BISI direct-signaling QUERY message is received at the TEO, the current busy/idle status is returned in either a BUSY or IDLE message. Using this mechanism, inconsistencies in busy/idle status between the NCP and the TEO can be corrected.

5.2.2 Activation and deactivation of busy/idle reporting

The TEO reports changes in busy/idle status to the NCP only if busy/idle reporting is activated for the SFG. Before busy/idle reporting can be activated for an SFG, common data between the NCP and the TEO must be verified, so that busy/idle reporting will function correctly. The data verification sequence is performed using direct-signaling messages and can be initiated from either the NCP or the TEO.

Under normal circumstances, the NCP controls the activation/deactivation process, but the TEO can also activate and deactivate busy/idle reporting for an SFG. As an example, when the TEO wishes to change some data (e.g., to add another line) associated with a particular activated 800 Service SFG, the SFG must first be deactivated. Next, a data verification process is initiated by the TEO. The NCP returns its data for that SFG, and the TEO verifies that the data returned agree with the data stored at the TEO. If so, the SFG is activated and busy/idle reporting resumes.

5.2.3 800 Service data for customers

The 800 Service customers can receive data concerning the activity on their lines. SFG attempt and overflow data are available using customer premises equipment. The counts can be sent to the customer as often as every 30 minutes. In addition, overflow counts are available on the customer's monthly bill.

Prior to the new System 800, all counts were kept solely at the TEO. With System 800, most overflows for customers served by TEOs with

the BISI feature occur at the NCP since the NCP screens all 800 Service calls.

To provide this overflow data to the customers, the NCP sends it to the TEO in a direct-signaling message either every 15 minutes or every day. The counts are then stored at the TEO so that the customer can continue to get an accurate picture of the activity on their 800 Service lines.

VI. SUMMARY

No. 1 ESS was the Bell System's first major commercial electronic switching system. Today No. 1 and 1A ESSs are an important part of the rapidly growing SPC network. The SPC network uses CCIS to provide rapid and efficient basic call handling. However, the network's full capabilities are just beginning to be realized now that the SPC network is being used for the development of services such as System 800. No. 1/1A ESSs play an important role in the SPC network because they provide direct interfaces to the subscribers and users of SPC network services. No. 1/1A ESSs serve nearly one-half of the Bell System subscriber lines. Thus, they have the potential to provide much customer-related service information and line status information to other SPC network switching offices and NCPs.

The No. 1/1A ESS uses a layered CCIS signaling subsystem that supports two types of data-link hardware at the innermost layer, while providing hardware-independent signaling capabilities to ESS applications software at the outer layers.

The CCIS call processing provides call handling over CCIS trunks in both local and toll offices. To do this, especially for local offices, the CCIS programs have to interface with numerous existing ESS call-processing programs which had previously used ingrained in-band signaling techniques. Changes to the ESS supervisory programs helped to provide new signaling interfaces to these programs.

The CCIS call processing must also respond to the many call-handling messages which are part of the CCIS protocol. It uses a finite-state machine program structure to respond to CCIS messages and, at the same time, to verify their reasonableness when received at different stages of a call.

No. 1/1A ESS performs two important System 800 functions. With the help of NCPs, it can screen 800 Service calls in either local or toll offices using its OSO feature. With OSO, System 800 NCPs are queried to obtain call-routing information for 800 Service calls. The OSO feature also provides code controls on 800 Service calls.

A System 800 BISI feature monitors System 800 customer line groups served by No. 1/1A ESS local offices and reports busy/idle line status changes to System 800 NCPs for use in call screening. The BISI feature

also provides 800 Service call attempt and overflow data to customer premises equipment. The System 800 features exemplify the types of SPC network and customer interfaces that No. 1/1A ESS can provide.

REFERENCES

1. J. J. Lawser, R. E. LeCronier, and R. L. Simms, "Stored Program Controlled Network: Generic Network Plan," B.S.T.J., *61*, No. 7 (September 1982), pp. 1589–98.
2. C. A. Dahlbom and J. S. Ryan, "Common-Channel Interoffice Signaling: History and Description of a New Signaling System," B.S.T.J., *57*, No. 2 (February 1978), pp. 225–50.
3. R. F. Frerking and M. A. McGrew, "Stored Program Controlled Network: Routing of Direct-Signaling Messages in the CCIS Network," B.S.T.J., *61*, No. 7 (September 1982), pp. 1599–1609.
4. D. J. Eigen and E. A. Youngs, "Stored Program Controlled Network: Calling Card Service—Human Factors Studies," B.S.T.J., *61*, No. 7 (September 1982), pp. 1715–35.
5. P. R. Miller and R. E. Wallace, "Common-Channel Interoffice Signaling: Signaling Network," B.S.T.J., *57*, No. 2 (February 1978), pp. 263–82.
6. L. M. Croxall and R. E. Stone, "Common-Channel Interoffice Signaling: No. 4 ESS Application," B.S.T.J., *57*, No. 2 (February 1978), pp. 367–72.
7. D. Sheinbein and R. P. Weber, "Stored Program Controlled Network: 800 Service Using SPC Network Capability," B.S.T.J., *61*, No. 7 (September 1982), pp. 1737–44.